On a Controller Parameterization for Infinite-Dimensional Feedback Systems Based on the Desired Overshoot

LIBOR PEKAŘ

Department of Automation and Control Engineering
Faculty of Applied Informatics
Tomas Bata University in Zlín
nam. T. G. Masaryka 5555, 76001 Zlín
CZECH REPUBLIC
pekar@fai.utb.cz   http://www.fai.utb.cz

Abstract: - The aim of this paper is to introduce, in detail, a novel approach for tuning of anisochronic single-input single-output controllers for infinite-dimensional feedback control systems. A class of Linear Time-Invariant Time Delay Systems (LTI TDSs) is taken as a typical representative of infinite-dimensional systems. Control design to obtain the eventual controller structure is made in the special ring of quasipolynomial meromorphic functions (R_{MS}). The use of this algebraic approach with a simple feedback loop for unstable or integrating systems leads to infinite-dimensional (delayed) controllers as well as the whole feedback loop. A natural task is to set tunable controller parameters in order to form the crucial area of the infinite closed-loop spectrum. It is worth noting that not only poles yet also zeros are taken into account. The prescribed positions of the right-most reference-to-output poles and zeros are given on the basis of the desired overshoot for a simple finite-dimensional matching model the detailed analysis of which is provided. The dominant poles and zeros are shifted to the prescribed positions using the Quasi-Continuous Shifting Algorithm (QCSA) followed by the use of an advanced optimization algorithm. The whole methodology is called the Pole-Placement Shifting based controller tuning Algorithm (PPSA). The PPSA is demonstrated on the setting of parameters of delayed controller for an unstable time delay plant of a skater on the controlled swaying bow. This example, however, shows a treachery of the algorithm and a natural feature of an infinite-dimensional system – namely, that its spectrum or even its dominant part can not be placed arbitrarily. Advantages and drawback as well as possible modification of the algorithm are also discussed.

Key-Words: - Infinite-dimensional systems, Time delay systems, Algebraic control design, Controller tuning, Pole-assignment, Pole-shifting, Desired overshoot, Optimization

1 Introduction

Infinite-dimensional systems constitute a huge class of complex systems usually expressed and mathematically formulated by Partial Differential Equations (PDAs). These systems and models are characterized by an infinite spectrum, i.e. with infinite many modes of the solution of and PDA. A family of Time Delay Systems (TDSs) stands for a quintessential representative of infinite-dimensional systems. Analytic solutions of many PDAs for systems with distributed parameters lead to a TDS in the form of Ordinary Differential Equations (ODEs) with deviated parameters or Difference-Differential Equations (DDEs), which ought to be – more precisely – include in the set of Functional Differential Equations (FDEs) [1] - [4], or a PDA can be equivalently expressed by convolutions or Riemann-Stieltjes integrals [5] - [6].

Linear time-invariant (LTI) TDSs, that are the matter of this contribution, are modelled in the state space by the linear FDEs which can be formulated using the Laplace transform in the form of transfer functions as well. However, these functions are no more rational but fractions of so-called quasipolynomials. Some authors, e.g. [7], pointed out that pointed out that the use of quasipolynomials does not permit to effectively handle some stabilization and control tasks, thus other rings based on quasipolynomials [8], [9] or its approximation [10], [11] for LTI TDS were introduced,. The ring of quasipolynomial meromorphic functions (R_{MS}), originally developed in [12] and revised and extended in [13], is another option. The development of the ring has been motivated by the endeavor not to loose dynamic information and the finding that Laplace and delays operators might not be seen as independent. Controller design in the ring employs the Bézout identity to obtain stable and proper controllers along with the Youla-Kučera parameterization to meet other control requirements.

In many cases, namely, for stable controlled plants or using a more advanced control system for even unstable ones [14], control design in R_{MS} ensures that the feedback loop is finite-dimensional in the sense that at least the reference-to-output transfer function (i.e. the complementary sensitivity function) has a finite number of poles. Note that even if the number of transfer function poles is finite, the number of system (characteristic) roots can be infinite – this is what we call "quasi-finite pole assignment". However, these latter poles are given by the dynamics of the controlled plant and their distribution can not be influenced by the feedback.

Hence, in other cases – especially for unstable LTI TDS, the control algorithm must deal with infinitely (countable) many feedback characteristic (transfer function) poles the positions of which depend on the selectable controller parameters. The use of pole-placement (pole-assignment, root-locus) tuning algorithms for LTI TDS can be a possible way how solve the problem, see e.g. [15] - [17]. However, these algorithms deal with poles only ignoring closed-loop zeros and they have been derived for state-space controllers.

In this paper, we concentrate on a procedure how to reach the desired dominant part of the spectrum and closed-loop zeros distribution as close as possible to a prescribed finite-dimensional model of the feedback control system. The idea is based on the analysis of a simple finite-dimensional model where the relative maximum overshoot, relative dumping and relative time-to-overshoot (i.e. a phase in some sense) are calculated and serve as a control performance indicators. Then, according to the selected values, the desired positions of dominant (i.e the rightmost) poles and zeros are calculated, and poles and zeros of a feedback system are shifted to the prescribed positions while the rest of the spectrum is pushed to the left (i.e. to the "stable" region). Moreover, the initial solution obtained using the Quasi-Continuous Shifting Algorithm (QCSA) [15], [16] is improved by an advanced numerical optimization algorithm, such as the Nelder-Mead (NM) algorithm [18], (Extended) Gradient Sampling Algorithm (EGSA) [19], [20] or the Self-Organizing Migration Algorithm (SOMA) [21]. The method is similar to that independently developed in [17]; however, essential differences are explained in this contribution.

The presented methodology is called Pole-Placement Shifting based controller tuning Algorithm (PPSA) and its basic ideas were formulated in [22]. A detailed analysis and a thorough simulation example extending the primordial study is provided in this contribution. Moreover, a different modification of the algorithm, which gives better results, is chosen here. The demonstrative example is dedicated to controller parameters tuning for an unstable TDS of a skater on the controlled swaying bow [23], where the controller is designed in an algebraic way using the R_{MS} ring. However, the obtained solution exposes an important feature of pole-placement methods for infinite-dimensional systems that it is not always possible to reach the desired spectrum even its finite part.

The paper is organized as follows. A concise introductory general description of LTI TDS, as a representative of infinite-dimensional systems, is presented in Section 2. Problem formulation and basic steps of the PPSA (and its modification utilized here) is the matter of Section 3. In Section 4, a detailed analysis of a selected finite-dimensional matching model is introduced. A complex Matlab-Simulink demonstrative example of control design of an unstable LTI TDS and its tuning using PPSA is presented in Section 5. Finally, Section 6 provides results analysis and discussion on the PPSA.

2 LTI TDS Description

Since the description of (LTI) TDS was the matter of many books, journal and conference contributions, a very short overview is given here.

A LTI TDS, no matter if a plant or a feedback loop, can be formulated by state and output FDEs in the following form [2], [24]

$$\begin{aligned} \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} &= \sum_{i=1}^{N_H} \mathbf{H}_i \frac{\mathrm{d}\mathbf{x}(t-\eta_i)}{\mathrm{d}t} + \mathbf{A}_0\mathbf{x}(t) + \sum_{i=1}^{N_A} \mathbf{A}_i \mathbf{x}(t-\eta_i) \\ &+ \mathbf{B}_0\mathbf{u}(t) + \sum_{i=1}^{N_B} \mathbf{B}_i \mathbf{u}(t-\eta_i) \\ &+ \int_0^L \left[\widetilde{\mathbf{A}}(\tau)\mathbf{x}(t-\tau) + \widetilde{\mathbf{B}}(\tau)\mathbf{u}(t-\tau)\right] \mathrm{d}\tau \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \int_0^L \widetilde{\mathbf{C}}(\tau)\mathbf{x}(t-\tau)\mathrm{d}\tau \end{aligned} \tag{1}$$

where $\mathbf{x} \in \mathrm{P}^n$ is a vector of state variables, $\mathbf{u} \in \mathrm{P}^m$ stands for a vector of inputs, $\mathbf{y} \in \mathrm{P}^l$ represents a vector of outputs, $\mathbf{A}_i$, $\widetilde{\mathbf{A}}(\tau)$, $\mathbf{B}_i$, $\widetilde{\mathbf{B}}(\tau)$, $\mathbf{C}$, $\widetilde{\mathbf{C}}(\tau)$, $\mathbf{H}_i$ are matrices of compatible dimensions, $0 \le \eta_i \le L$ are *lumped* (discrete) delays and convolution integrals express *distributed* delays. If $\mathbf{H}_i \neq \mathbf{0}$ for any $i = 1,2,...N_H$, model (1) is called *neutral*; on the other hand, if $\mathbf{H}_i = \mathbf{0}$ for every $i = 1,2,...N_H$, a so-called *retarded* model is obtained.

Another, operator-based description of LTI TDS has been introduced in [25], [26]. Some other models can be found in a brilliant overview [24].

Considering model (1) and zero initial conditions, a general multi-input multi-output (MIMO) system in the form of the following transfer matrix is obtained

$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) = \frac{\mathbf{C}(s)\operatorname{adj}[s\mathbf{I} - \mathbf{A}(s)]\mathbf{B}(s)}{\det[s\mathbf{I} - \mathbf{A}(s)]}\mathbf{U}(s)$$

$$\begin{aligned}\mathbf{A}(s) &= s\sum_{i=1}^{N_H} \mathbf{H}_i \exp(-s\eta_i) + \mathbf{A}_0 + \sum_{i=1}^{N_A} \mathbf{A}_i \exp(-s\eta_i) \\ &\quad + \int_0^L \widetilde{\mathbf{A}}(\tau)\exp(-s\tau)\mathrm{d}\tau \end{aligned} \tag{2}$$

$$\mathbf{B}(s) = \mathbf{B}_0 + \sum_{i=1}^{N_B} \mathbf{B}_i \exp(-s\eta_i) + \int_0^L \widetilde{\mathbf{B}}(\tau)\exp(-s\tau)\mathrm{d}\tau$$

$$\mathbf{C}(s) = \mathbf{C} + \int_0^L \widetilde{\mathbf{C}}(\tau)\exp(-s\tau)\mathrm{d}\tau$$

All transfer functions in $\mathbf{G}(s)$ - or a transfer function in a single-input single output (SISO) case - have the identical denominator in the form

$$\begin{aligned} m(s) &= \operatorname{num}\det[s\mathbf{I} - \mathbf{A}(s)] = \operatorname{num}M(s) \\ &= s^n + \sum_{i=0}^{n}\sum_{j=1}^{h_i} m_{ij}s^i \exp(-s\eta_{ij}), \eta_{ij} \ge 0 \end{aligned} \tag{3}$$

which is called the *characteristic quasipolynomial* of the system.

Poles $\sigma_i, i \in \mathrm{N}$ of TDS are solutions of the equation $M(s) = \det[s\mathbf{I} - \mathbf{A}(s)] = 0$, whereas zeros $\zeta_i, i = 1,2,...\infty$, in a SISO case are given by the solution of $G(s) = M^{-1}(s)\mathbf{C}(s)\operatorname{adj}[s\mathbf{I} - \mathbf{A}(s)]\mathbf{B}(s) = 0$. Due to transcendental character of $M(s)$ caused by functionality of its exponential terms, the number of poles is infinite.

Asymptotic stability agrees with that notion known from finite-dimensional systems, i.e. all poles are to be located in the open left-half plane, X_0^-.

3 Problem Formulation and PPSA

Many control design procedures for (LTI) TDS - principally, those except the class of Finite Spectrum Assignment resulting in extremely complex control law - yield the infinite-dimensional feedback as well. The eventual controller(s) naturally include(s) tunable parameters which have to be appropriately set. The controller design procedure in the R_{MS} ring, used in this paper, gives rise to a set $\mathbf{K}$ of mostly bounded real-valued parameters. In the light of the asymptotic stability requirement, it is necessary to get the feedback spectrum in X_0^-, which means to obtain the so-called spectral abscissa

$$\alpha_P(\mathbf{K}) := \max \operatorname{Re}\sigma_i;\ \sigma_i \in \Omega_P : M(\sigma_i) = 0 \tag{4}$$

strictly negative, as solved e.g. in [15].

However, we are going beyond this basic claim, in this paper. A sub-optimal controller tuning idea based on the desired or ultimate position of the right-most feedback poles is presented. The algorithm stems from the dependence of the maximum relative step response overshoot, the relative dumping factor and the relative time-to-overshoot on the position of poles and zeros of a finite-dimensional model. Thus, the goal is to match the dominant (i.e. the right-most) region of the feedback spectrum with that of the desired finite-dimensional model. The procedure has been called PPSA and its basic steps follows. As first, the original scheme presented in [22] is provided (version 1); then, as second, its modification for a limit case is suggested (version 2). Finally, the version utilized in this paper (version 3) is given to the reader.

3.1 PPSA (version 1)

Input: Closed-loop reference-to-output transfer function $G_{WY}(s)$ with the number of r selectable (tunable) parameters in the set

$$\mathbf{K} = \{k_1, k_2, ..., k_r\} = \mathbf{K}_{num} \cup \mathbf{K}_{den} \tag{4}$$

where $\mathbf{K}_{num} \subseteq \mathbf{K}, |\mathbf{K}_{num}| = r_{num}$ are parameters in the numerator, whereas $\mathbf{K}_{den} \subseteq \mathbf{K}, |\mathbf{K}_{den}| = r_{den}$ means selectable parameters in the denominator and let $r_{nd} = |\mathbf{K}_{num} \setminus (\mathbf{K}_{num} \cap \mathbf{K}_{den})| = |\mathbf{K}_{nd}|$.

Step 1: Choose a simple model of a stable finite-dimensional system with the unit static gain with the transfer function $G_{WY,m}(s)$ with the numerator of degree $n_{num} \leq r_{nd}$ and the denominator of degree $n_{den} \leq r_{den}, n_{num} < n_{den}$. Calculate step response maximum overshoots, relative dumping factors and relative times-to-overshoot of the model as performance measures for a suitable range of its n_{num} zeros and n_{den} poles (including their multiplicities).

Step 2: Prescribe all poles of the model with respect to the calculated performance factors and place the number of n_{den} closed-loop poles of the system to these desired positions. If the placed poles are dominant (i.e. the rightmost), go to Step 6; otherwise, go to Step 3.

Step 3: Initialize the counter of currently shifted poles as $n_{sp} = n_{den}$.

Step 4: Shift the rightmost feedback system poles towards the prescribed locations successively using the QCSA [15], [16]. If necessary, increase n_{sp}. If $n_{den} < n_{sp} \leq r_{den}$, try to move the rest of dominant (rightmost) poles to the left, again e.g. using QCSA.

Step 5: If all prescribed poles are dominant, the procedure is finished. Otherwise, select a suitable cost function reflecting the distance of dominant poles from prescribed positions and the spectral abscissa. Minimize the cost function using an advanced (genetic, direct-search, etc.) iterative algorithm, e.g. see [18] - [21].

Step 6: Do Steps 3-5 for prescribed zeros, where it holds for the number n_{sz} of currently shifted zeros that $n_{num} \leq n_{sz} \leq r_{nd}$, to update the values of $\mathbf{K}_{nd}$.

Output: The vector of controller parameters $\mathbf{K}$ and positions of the rightmost poles and zeros.

The presented original version of the PPSA prefers the positions of feedback poles at the expense of zeros since poles are placed primarily due to their more significant affect to the dynamics. Once the set $\mathbf{K}_{den}$ is found, these values are fixed in the numerator and $\mathbf{K}_{nd}$ is to be found subsequently. In the limit case $r_{nd} = 0$, positions of feedback system zeros can not be influenced at all and Step 1 of the algorithm allows to have only a constant numerator of $G_{WY,m}(s)$. To be more flexible, if it holds that $r_{num} > 0$, the following modification of the PPSA can be performed.

3.2 PPSA (version 2)

Input: See the PPSA version 1.

Step 1: If $r_{nd} = 0$, $r_{num} > 0$, choose a simple model of a stable finite-dimensional system with the unit static gain with the transfer function $G_{WY,m}(s)$ with the numerator of degree $n_{num} \leq r_{num}$ and the denominator of degree $n_{den} \leq r - n_{num} = r_{den} - n_{num}$ and, moreover, $n_{num} < n_{den} \Leftrightarrow n_{num} < r/2$. Calculate step response maximum overshoots, relative dumping factors and relative times-to-overshoot of the model as performance measures for a suitable range of its n_{num} zeros and n_{den} poles.

Steps 2: Fix the number of n_{num} controller parameters so that zeros of $G_{WY}(s)$ are prescribed exactly. Hence, other parameters from $\mathbf{K}$ are dependent on these fixed ones.

Step 3-6: See Steps 2-5 of the PPSA version 1.

Output: The vector of controller parameters $\mathbf{K}$ and positions of the rightmost poles.

Note that there is not guaranteed that zeros are dominant in this version, yet they are placed exactly. Therefore, if at least holds that $r_{num} > 0$, the third version of the PPSA is suggested.

3.3 PPSA (version 3)

Input: Closed-loop reference-to-output transfer function $G_{WY}(s)$ with the number of r selectable (tunable) parameters in the set $\mathbf{K} = \mathbf{K}_{num} \cup \mathbf{K}_{den}$ where $\mathbf{K}_{num} \subseteq \mathbf{K}, |\mathbf{K}_{num}| = r_{num}$ are parameters in the numerator, whereas $\mathbf{K}_{den} \subseteq \mathbf{K}, |\mathbf{K}_{den}| = r_{den}$.

Step 1: If $r_{num} > 0$, choose a simple model of a stable finite-dimensional system with the numerator of degree n_{num}, the denominator of degree n_{den} and unit static gain governed by the transfer function $G_{WY,m}(s)$ where $n_{num} < n_{den}, n_{num} + n_{den} \leq r$.

Step 2: Prescribe all poles and zeros of $G_{WY,m}(s)$ with respect to the calculated performance factors and place the number of n_{den} closed-loop poles and the number of n_{num} zeros of the system to these desired positions. If the placed positions are dominant, the algorithm is finished; otherwise, go to Step 3.

Step 3: Initialize counters of currently shifted poles as $n_{sp} = n_{den}$ and zeros as $n_{sz} = n_{num}$.

Step 4: Shift the rightmost feedback system poles and zeros towards the prescribed locations. If necessary, increase n_{sp} and/or n_{sz}. If $n_{den} < n_{sp} \leq r_{den}$ or $n_{num} < n_{sz} \leq r_{num}$ try to move the rest of dominant poles or zeros, respectively, to the left.

Step 5: If all prescribed poles and zeros are dominant, the procedure is finished. Otherwise, select a suitable cost function reflecting the distance of dominant poles and zeros from prescribed positions and the (spectral) abscissa of both, poles and zeros. Minimize the cost function.

Output: The vector of controller parameters **K** and positions of the rightmost poles and zeros.

This version does not guarantee that either positions of the rightmost system poles and zeros are dominant, or they are placed exactly; however, poles are not preferred to zeros. Nevertheless, different weights on poles and zeros can be included in the definition of the cost function (see Step 5).

Note that (LTI) TDS of neutral type [3] requires including the restriction to so-called strong stability in the cost function [20].

4 Selected Model Analysis

The tuning algorithm presented in the previous subsection stems from the dependence of the maximum relative step response overshoot, the relative dumping and the relative time-to-overshoot (phase lag) on the position of poles and zeros of a desired finite-dimensional model. The methodology will be demonstrated on a second order model.

Hence, let the prescribed (desired) closed-loop model be

$$G_{WY,m}(s)=\frac{b_1 s+b_0}{s^2+a_1 s+a_0}=k\frac{s-z_1}{(s-s_1)(s-\bar{s}_1)} \qquad (5)$$

where $k,b_1,b_0,a_1,a_0\neq 0\in \mathrm{P}$ are model parameters $z_1\in \mathrm{P}^-$ stands for a model zero and $s_1=\alpha+\mathrm{j}\omega\in \mathrm{X}_0^-$ $,\alpha<0,\omega\geq 0$, is a model stable pole where $\bar{s}_1$ expresses its complex conjugate.

To obtain the unit static gain of $G_{WY,m}(s)$ it must hold that

$$\frac{b_0}{a_0}=1, k=-\frac{|s_1|^2}{z_1} \qquad (6)$$

Calculate the model impulse function $g_{WY,m}(t)$ of $G_{WY,m}(s)$ as

$$g_{WY,m}(t)=k\exp(\alpha t)\left[\cos(\omega t)-\frac{z_1-\alpha}{\omega}\sin(\omega t)\right] \qquad (7)$$

Since $i_{WY,m}(t)=h'_{WY,m}(t)$, where $h_{WY,m}(t)$ is the step response function, the necessary condition for the existence of a step response overshoot at time t_O is

$$i_{WY,m}(t_O)=0, t_O>0 \qquad (8)$$

Condition (8) yields these two solutions: either $t_O\to-\infty$ (which is trivial) or

$$t_O=\frac{1}{\omega}\arccos\left(\pm\frac{|\alpha-z_1|}{\sqrt{(\alpha-z_1)^2+\omega^2}}\right)=\frac{1}{\omega}\arctan\left(\frac{\omega}{-\alpha+z_1}\right) \qquad (9)$$

when considering $\arccos(\cdot),\arctan(\cdot)\in[0,\pi],\omega>0$. Obviously, (9) has infinitely many solutions. If $\alpha<0,z_1<0$, the maximum overshoot occurs at time

$$t_{\max}=\min(t_O) \qquad (10)$$

One can further calculate the step response function $h_{WY,m}(t)$ as

$$h_{WY,m}(t)=\frac{k}{|s_1|^2}\left[\exp(\alpha t)\left(z_1\cos(\omega t)-\frac{z_1\alpha-|s_1|^2}{\omega}\sin(\omega t)\right)-z_1\right] \qquad (11)$$

Define now the maximum relative overshoot as

$$\Delta h_{WY,m,\max}:=\frac{h_{WY,m}(t_{\max})-h_{WY,m}(\infty)}{h_{WY,m}(\infty)} \qquad (12)$$

see Fig. 1.

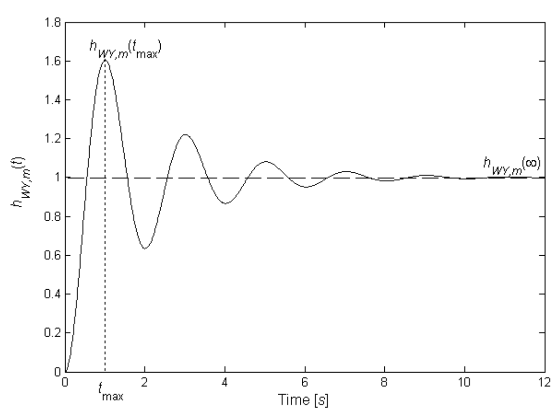

Fig. 1. Reference-to-output step response characteristics and the maximum overshoot

The overshoot then reads

$$\Delta h_{WY,m,\max} =$$
$$= \exp(\alpha t_{\max})\left(\frac{-z_1\omega\cos(\omega t_{\max}) + \left(z_1\alpha - |s_1|^2\right)\sin(\omega t_{\max})}{z_1\omega}\right)$$
$$= \frac{1}{\xi_z}\exp(-\xi_\alpha t_{\max,norm})\begin{pmatrix}-\xi_z\cos(t_{\max,norm}) \\ +\left(\xi_\alpha^2 + 1 - \xi_\alpha\xi_z\right)\sin(t_{\max,norm})\end{pmatrix} \quad (13)$$

where

$$t_O = \frac{1}{\omega}\arccos\left(\pm\frac{|\alpha - z_1|}{\sqrt{(\alpha - z_1)^2 + \omega^2}}\right) = \frac{1}{\omega}\arctan\left(\frac{\omega}{-\alpha + z_1}\right) \quad (14)$$
$$t_{\max} = \min(t_O)$$

and the normalized maximum-overshoot time (i.e. a phase lag) is

$$t_{\max,norm} = \omega t_{\max} = \min\left(\arccos\left(\pm\frac{|\xi_\alpha - \xi_z|}{\sqrt{(\xi_\alpha - \xi_z)^2 + 1}}\right)\right) = \min\left(\arctan\left(\frac{1}{\xi_\alpha - \xi_z}\right)\right) \quad (15)$$

and

$$\xi_\alpha = -\frac{\alpha}{\omega},\ \xi_z = -\frac{z_1}{\omega} \quad (16)$$

Note that $t_{\max,norm}$ has the meaning of a phase delay.

We can successfully use e.g. Matlab to display functions $\Delta h_{WY,m,\max}(\xi_\alpha,\xi_z)$ and $t_{\max,norm}(\xi_\alpha,\xi_z)$ graphically, for suitable ranges of ξ_α,ξ_z as can be seen, for instance, from Figs. 2 and 3.

To sum up, a user chooses the values of $\Delta h_{WY,m,\max}$, ξ_α (i.e. a relative dumping) and $t_{\max}$. Consequently, ξ_z is calculated from (13) and (15), which gives a triplet ω,α,z_1 from (16). Example given in Chapter 5 elucidates the procedure.

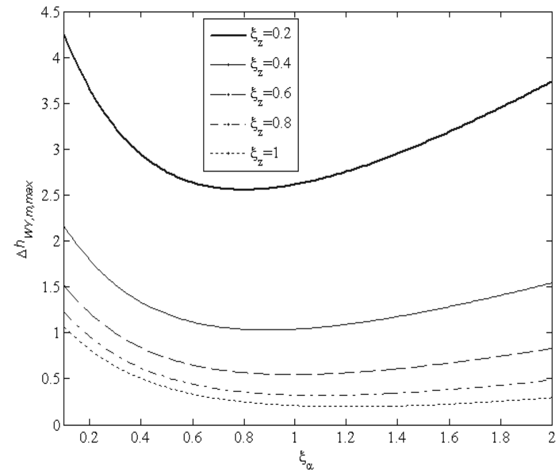

Fig. 2. Maximum overshoots $\Delta h_{WY,m,\max}(\xi_\alpha,\xi_z)$ for $\xi_\alpha = [0.1,2]$, $\xi_z = \{0.2,0.4,0.6,0.8,1\}$

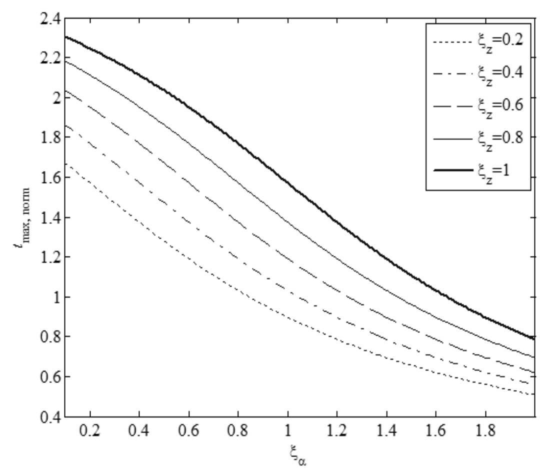

Fig. 3. Normalized maximum-overshoot times (phase lags) $t_{\max,norm}(\xi_\alpha,\xi_z)$ for $\xi_\alpha = [0.1,2]$, $\xi_z = \{0.2,0.4,0.6,0.8,1\}$

5 Demonstrative Example

The presented example demonstrates controller parameters tuning using the PPSA with the algebraic controller design in the R_{MS} ring for an unstable retarded LTI TDS plant.

Consider an unstable system as in Fig. 4 expressing a roller skater the controlled swaying bow. It has been stated in [23] that the transfer function of the system reads

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b\exp(-(\tau+\vartheta)s)}{s^2\left(s^2 - a\exp(-\vartheta s)\right)} \quad (17)$$

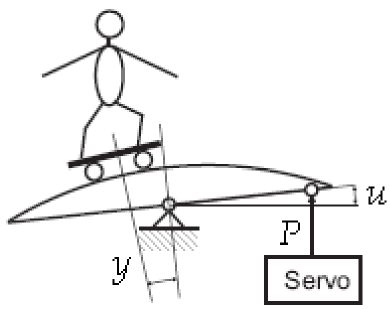

Fig. 4. Roller skater on the controlled swaying bow

In the model, $y(t)$ means the skater's deviation from the desired position, $u(t)$ expresses the slope angle of a bow caused by force the external force, delays τ, ϑ are the skater's and servo latencies, respectively, and b, a stand for positive real parameters. Skater controls the servo driving by remote signals into servo electronics. As presented in the literature, let b = 0.2, a = 1, $\tau = 0.3$ s, $\vartheta = 0.1$ s.

5.1 Controller structure design

Consider the well-know simple negative feedback loop with the controller given by the transfer function $G_R(s) = Q(s)/P(s)$. The first stage of the algebraic controller design (in the broader meaning) is the determination of the controller structure by plant transfer function factorization, calculation of all stabilizing controllers, parameterization of their structures etc. Since this task is not the aim of this paper, the reader is referred to [27] for details.

As a result of this stage, the following controller transfer function can be obtained

$$G_R(s) = \frac{Q(s)}{P(s)} = \frac{b\left(q_3 s^3 + q_2 s^2 + q_1 s + q_0\right)(s+m_0)^4 + p_0 m_0^4 s^2\left(s^2 - a\exp(-\vartheta s)\right)}{b\left[\left(s^3 + p_2 s^2 + p_1 s + p_0\right)(s+m_0)^4 - p_0 m_0^4 \exp(-(\tau+\vartheta)s)\right]} \tag{18}$$

Then the reference-to-output transfer function is given by (19). Notice that the characteristic quasipolynomial (i.e. the denominator of $G_{WY}(s)$) has two factors – a quasipolynomial and a polynomial one. Since the placement of a multiple (quadruple) real pole is trivial, we can concentrate on the quasipolynomial factor $g_{DEN}(s)$ with infinite number of its roots and seven selectable controller parameters, i.e. $p_2, p_1, p_0, q_3, q_2, q_1, q_0 \in \mathrm{P}$.

$$\begin{aligned} G_{WY}(s) &= \frac{1}{(s+m_0)^4}\frac{g_{NUM}(s)}{g_{DEN}(s)}\exp(-(\tau+\vartheta)s) \\ g_{NUM}(s) &= b\begin{pmatrix} \left(q_3 s^3 + q_2 s^2 + q_1 s + q_0\right)(s+m_0)^4 \\ + p_0 m_0^4 s^2\left(s^2 - a\exp(-\vartheta s)\right)\end{pmatrix} \\ g_{DEN}(s) &= s^2\left(s^2 - a\exp(-\vartheta s)\right)\left(s^3 + p_2 s^2 + p_1 s + p_0\right) \\ &\quad + b\exp(-(\tau+\vartheta)s)\left(q_3 s^3 + q_2 s^2 + q_1 s + q_0\right) \end{aligned} \tag{19}$$

5.2 Application of PPSA for controller parameterization

It is obvious from (19) that

$$\begin{aligned} \mathbf{K} &= [q_3, q_2, q_1, q_0, p_2, p_1, p_0]^T \\ r &= r_{den} = 7, r_{num} = 5, r_{nd} = 0 \end{aligned} \tag{20}$$

Since $r_{nd} = 0$, i.e. there is no free parameter in the numerator of (19) that is not included in the denominator, version 1 of the PPSA can no be used with model (5). This version allows only having zero-order numerator. The use of version 2 was the issue of paper [22]. Therefore, in this paper, we decided to test and verify version 3 of the PPSA, i.e. simultaneous shifting of feedback poles and zeros.

Recall that there are two factors in the characteristic quasipolynomial of the feedback system. Thus, to cancel the impact of the quadruple real pole $s_1 = -m_0$, it must hold that $m_0 >> -\alpha_P(\mathbf{K})$.

Let us make the following option: $\Delta h_{WY,m,\max} = 0.5$, $\xi_\alpha = 0.5$ and $t_{\max} = 10$ s. Note that it is not apriori guaranteed that the desired performance measures will be meet. From Fig. 2 we have $\xi_z = 0.9$. By taking consideration of these values, Fig. 3 gives rise to $t_{\max,norm} \approx 2$ which results in $\omega = 0.2, z_1 = -0.18, \alpha = -0.1$. Since $\alpha P(\mathbf{K}) \approx \alpha$, choose $m_0 = 5$.

Initial direct pole placement, see e.g. [6], yileds controller parameters

$$\mathbf{K}_0 = \begin{bmatrix} 1.1014, 0.9852, -0.0113, 0.0171, \\ 1.113, 0.7, 0.2411 \end{bmatrix}^T \tag{21}$$

which gives the right-most spectrum of poles

$$\Omega_{P,0} = \begin{Bmatrix} 0.8959, 0.1445, -0.1 \pm 0.2\mathrm{j}, \\ -0.5201 \pm 0.5029\mathrm{j}, -1.0114 \end{Bmatrix} \tag{22}$$

and zeros

$$\Omega_{Z,0} = \begin{Bmatrix} -0.1373 \pm 0.1536\mathrm{j}, -0.18, \\ -1.0822, -2.3507 \pm 0.8.4523\mathrm{j} \end{Bmatrix} \tag{23}$$

Obviously, the prescribed poles and zeros are not dominant ones. The process of the PPSA is described by the evolution of controller parameters **K**, the spectral abscissa $\alpha_P(\mathbf{K})$ (i.e. the real part of the rightmost pole pair $\sigma_1, \overline{\sigma}_1$), the abscissa of zeros, $\alpha_Z(\mathbf{K})$, defined as

$$\alpha_Z(\mathbf{K}) := \max \operatorname{Re} \zeta_i;\ \zeta_i \in \Omega_Z : G_{WY}(\zeta_i) = 0 \tag{24}$$

the distance of the dominant pole from the prescribed one, $|\sigma_1 - s_1|$, and that of the dominant zero, ζ_1, from the prescribed one $|\zeta_1 - z_1|$, which is displayed in Figs. 5 – 9, respectively.

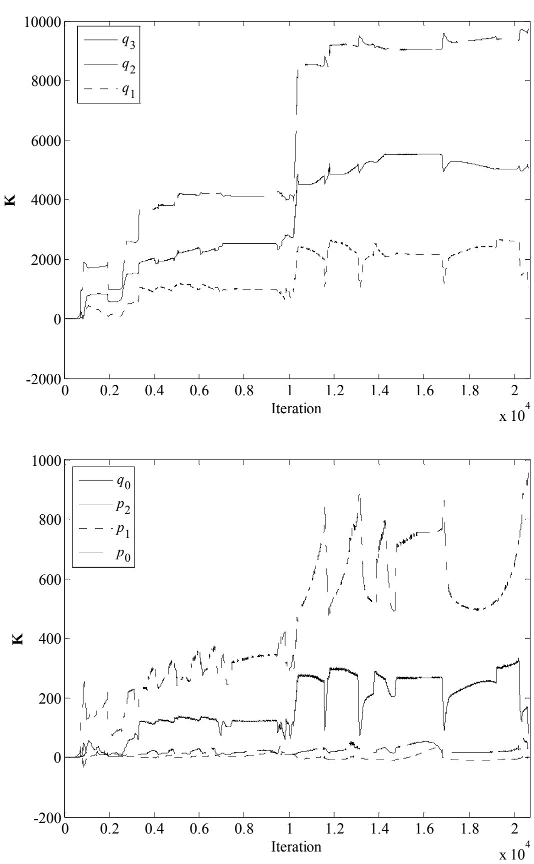

Fig. 5. Evolution of **K** using the PPSA (Steps 1-4)

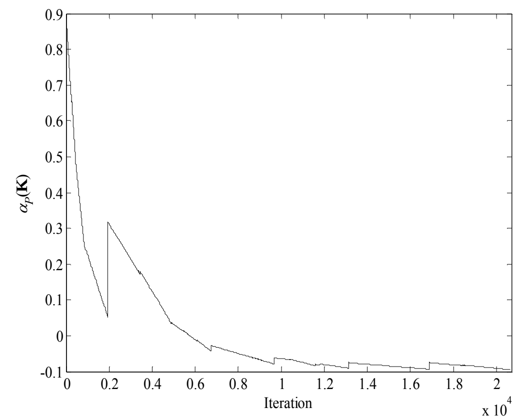

Fig. 6. Evolution of $\alpha_P(\mathbf{K})$ using the PPSA (Steps 1-4)

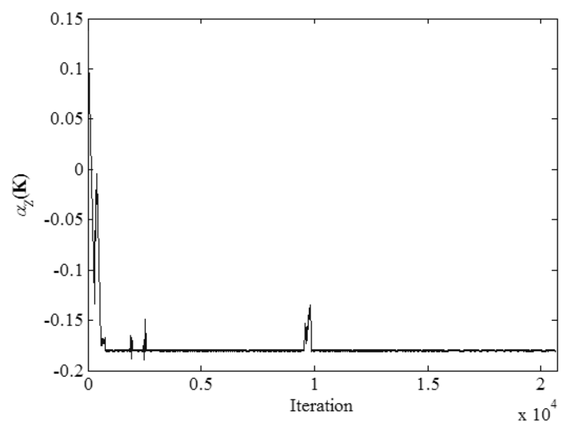

Fig. 7. Evolution of $\alpha_Z(\mathbf{K})$ using the PPSA (Steps 1-4)

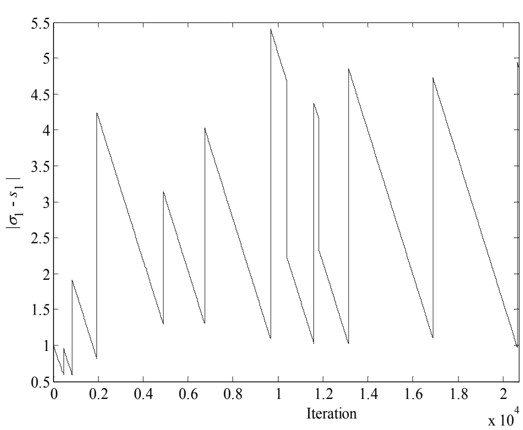

Fig. 8. Evolution of $|\sigma_1 - s_1|$ using the PPSA (Steps 1-4)

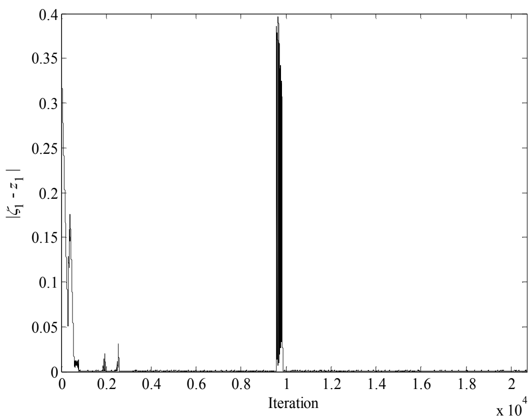

Fig. 9. Evolution of $|\zeta_1 - z_1|$ using the PPSA (Steps 1-4)

Eventual controller parameters obtained from the use of the QCSA in the PPSA are

$$\mathbf{K}_{20636} = \begin{bmatrix} 5051.788, 9734.946, 1046.105, 278.9573, \\ 32.3117, 1.7838, 954.866 \end{bmatrix}^T \tag{25}$$

The obtained spectra read

$$\Omega_{P,20636} = \begin{Bmatrix} 0.00945 \pm 1.1778\mathrm{j}, -0.1168 \pm 0.0697\mathrm{j} \\ -0.118 \pm 5.0275\mathrm{j}, \end{Bmatrix} \tag{26}$$

and

$$\Omega_{Z,20636} = \begin{Bmatrix} -0.1804, -0.22 \pm 0.1187\mathrm{j}, -0.7546, \\ -2.7809 \pm 8.2997\mathrm{j} \end{Bmatrix} \tag{27}$$

Obviously, the algorithm tried to keep the rightmost zero as close to the prescribed one as possible, while to shift the rightmost pole. However, the distance is cyclically changed so that there is not possible to get closer without exceeding values of controller parameters.

Step 5 of the algorithm described in Subchapter 3.3 follows, in order to improve the obtained result. Let us define the objective function

$$\Phi(\mathbf{K}) = |\sigma_1 - s_1| + |\zeta_1 - z_1| + \lambda_1 \alpha_{r,P}(\mathbf{K}) + \lambda_2 \alpha_{r,Z}(\mathbf{K}) \tag{28}$$

where $\alpha_{r,P}(\mathbf{K})$ means the spectral abscissa of the rest of poles except the dominant pair, $\alpha_{r,Z}(\mathbf{K})$ has the same meaning yet for zeros, and λ_1, λ_2 are weighting parameters.

For the minimization of (28), the SOMA [21] belonging to genetic algorithms has been chosen. If $\lambda_1 = \lambda_2 = 0.01$, results are then the following

$$\begin{aligned} \mathbf{K}_{20636,opt} &= \begin{bmatrix} 5235.169, 9829.219, 1060.87, 78.2405, \\ 30.9684, 1.763, 947.517 \end{bmatrix}^T \\ \Omega_{P,20636,opt} &= \begin{Bmatrix} -0.1158 \pm 0.0674\mathrm{j}, -0.1161 \pm 5.1163\mathrm{j}, \\ -0.1211 \pm 1.2103\mathrm{j} \end{Bmatrix} \\ \Omega_{Z,20636,opt} &= \begin{Bmatrix} -0.1801, -0.2247 \pm 0.1032\mathrm{j}, -0.7607, \\ -2.817 \pm 8.1939\mathrm{j} \end{Bmatrix} \end{aligned} \tag{29}$$

As can be seen, the obtained results do not significantly differ from the ones introduced in (25) – (27). However, unfortunately, final poles and zeros positions are quite far from the desired ones, which has decisive impact to feedback dynamic properties and proves the fact about (LTI) TDS that the desired spectrum can not be chosen arbitrarily but the “energy” of the system has to be taken into consideration. Here, the proposed controller structure allows only a poor stabilization (or pole-zero assignment) for the unstable controlled plant.

6 Discussion

Some aspects of the proposed PPSA methodology and particularly, its version have to be mentioned.

As first, in contrast to a similar idea independently introduced [17], there are some differences in the PPSA. Firstly, the PPSA uses the input-output space of meromorphic Laplace transfer functions, whereas the one in [17] deals purely with the state space. Moreover, poles are initially placed in desired positions unambiguously according to the estimated maximal overshoot; however, they can leave their positions during the shifting. In [17], the can not leave the prescribed positions which may yield to a lengthy trial-and-reset procedure. Here, in the PPSA, the dominant poles move to the prescribed ones and the rest of the spectrum is pushed to the left again by minimization of an objective function (including the spectral abscissa), without the requirement of resetting the selection of assigned poles. Last but not least, an optimization algorithm (SOMA) is utilized as a minimization technique, instead of a simple shifting by the QCSA or the HANSO algorithm used in [17].

As second, the initial shifting may be improved by the use of other “approaching” strategies. In the current versions, the dominant closed-loop poles and zeros are shifted to the rightmost prescribed ones in such way that a complex conjugate pair is considered as a one point of attraction. In other words, a real pole moves to a prescribed pole from a complex conjugate pair and viceversa. It is not problem if the convergence is sufficient since, at the end, a pair "prescribed-true" pole (zero) is of the same type, i.e. "real-real" or "complex-complex". However, the convergence and speed of the PPSA might be improved by the strategy that only poles (zeros) of the same type (real, complex) are approaching to each other, or by thorough consideration that a complex conjugate pair means two separate roots instead one.

As third, better optimization procedures can be utilized in Step 5 of the PPSA (version 3), e.g. the well-known and efficient NM algorithm [18]. In fact, the SOMA requires a quantity of cost function (zero/pole location) calculations, in the worst case we have $\mathrm{round}\left(\left(PopSize-1\right)\cdot PathLength/Step\right)$ spectrum enumerations in an iterative step where $PopSize$ is the size of the population, $PathLength$ agrees with the path length on the way to the leader and $Step$ means the discretization step; see details in [21]. In contrast to that, the NM algorithm needs only (at most) $r+1$ spectrum evolutions.

There are also other arguable topics about the PPSA, as another evaluation of pole (zero) dominancy, selection of another cost function etc. These problems and task can be touched or even solved in the further research.

6 Conclusion

The presented paper has introduced a novel version of an original iterative algorithm for pole-assignment and spectral optimization of infinite-dimensional systems (LTI TDS) by shifting of dominant poles and zeros to the prescribed positions. The method is based on the desired feedback step response overshoot, the relative dumping factor and a phase lag based on a selected finite-dimensional model. The eventual controller structure with real selectable parameters has been designed in the recently revised ring of meromorphic functions. The subsequent example has demonstrated the usability of the method, yet, it has pointed out its weaknesses. Some ideas in the last section of this contribution have suggested ways how the results can be improved and extended. It gives rise to future efforts to enhance the methodology.

Acknowledgements

The author kindly appreciates the financial support which was provided by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References:

[1] R. Bellman and K.L. Cooke, *Differential-Difference Equations*, New York: Academic Press, 1963.

[2] H. Górecki, S. Fuksa, P. Grabowski and A. Korytowski, *Analysis and Synthesis of Time Delay Systems*, Warszawa: PWN – Polish Scientific Publishers, co-editor Wiley, 1989.

[3] J.K. Hale and S.M. Verduyn Lunel, *Introduction to Functional Differential Equations*, in *Applied Math. Sciences*, Vol.99, New York: Springer-Verlag, 1993.

[4] V.B. Kolmanovskii and A. Myshkis, *Introduction to the Theory and Applications of Functional Differential Equations*, Dordrecht: Kluwer Academic Publishers, 1999.

[5] P. Zítek, Anisochronic state theory of dynamic systems, *Acta Technica ČSAV*, Vol. 4, 1983.

[6] P. Zítek and A. Víteček, *Control of Systems with Delays and Nonlinearities* (in Czech), Prague: ČVUT Publishing, 1999.

[7] D. Brethé and J.J. Loiseau, An Effective Algorithm for Finite Spectrum Assignment of Single-Input Systems with Delays, *Mathematics and Computers in Simulation*, Vol.45, No.3-4, 1998, pp. 339-348.

[8] H. Gluesing-Lueerssen, A Behavioral Approach to Delay-Differential Systems, *SIAM Journal of Control and Optimization*, Vol.35, No.2, 1997, pp. 480-499.

[9] J.J. Loiseau, Algebraic Tools for the Control and Stabilization of Time-Delay Systems, *Annual Reviews in Control*, Vol.24, 2000, pp. 135-149.

[10] P. Dostál, V. Bobál, Z. Babík, Control of Unstable and Integrating Time Delay Systems using Time Delay Approximations, *WSEAS Transactions on Systems*, Vol.11, No. 10, 2012, pp. 586-595.

[11] P. Dostál, F. Gazdoš, V. Bobál, Design of Controllers for Time Delay Systems. Part II. Integrating and Unstable Systems, *Journal of Electrical Engineering*, Vol.59, No.1, 2008, pp. 3-8.

[12] P. Zítek and V. Kučera, Algebraic Design of Anisochronic Controllers for Time Delay Systems, *International Journal of Control*, Vol.76, No.16, 2003, pp. 1654-1665.

[13] L. Pekař, A Ring for Description and Control of Time-Delay Systems, *WSEAS Transactions on Systems, Special Issue: Modelling, Identification, Stability, Control and Applications*, Vol.11, No.10, 2012, pp. 571-585.

[14] L. Pekař, J. Korbel and R. Prokop, TFC Robust Control Design of Time Delay Systems – Analysis and Example", in *Proceedings of the 19th International Conference on Process Control* (*PC 2013*), Štrbské Pleso, Slovak Republic, 2013. Accepted.

[15] W. Michiels, K. Engelborghs, P. Vansevevant, and D. Roose, Continuous Pole Placement for Delay Equations, *Automatica*, Vol.38, No.5, pp. 747-761, 2002.

[16] W. Michiels and T. Vyhlídal, An Eigenvalue Based Approach for the Stabilization of Linear Time-Delay Systems of Neutral Type, *Automatica*, Vol.41, No.6, 2005, pp. 991-998.

[17] W. Michiels, T. Vyhlídal, and P. Zítek, Control Design for Time-Delay Systems Based on Quasi-Direct Pole Placement, *Journal of Process Control*, Vol.20, No.3, 2010, pp. 337-343.

[18] J.A. Nelder and R. Mead, A Simplex Method for Function Minimization, *The Computer Journal*, Vol.7, No.4, 1965, pp. 308–313.

[19] T. Vanbiervliet, K. Verheyden, W. Michiels, and S. Vandewalle, A Nonsmooth Optimization Approach for the Stabilization of Time-Delay Systems, *ESIAM: Control, Optimisation and Calculus of Variations*, Vol.14, No.3, 2008, pp. 478-493.

[20] T. Vyhlídal, W. Michiels and P. McGahan, Synthesis of a Strongly Stable State-Derivative Controller for a Time Delay System Using Constrained Nonsmooth Optimization, *IMA Journal of Mathematical Control and Information*, Vol.27, No.4, 2010, pp. 437-455.

[21] I. Zelinka, SOMA-Self Organizing Migrating Algorithm, in *New Optimization Techniques in Engineering*, G. C. Onwobolu, B. V. Babu (eds.), pp. 167-217, Springer, Berlin, 2004.

[22] L. Pekař, PPSA: Root-Locus Tuning Technique for Infinite-Dimensional Systems, in *Computational Methods in Science & Engineering*. *Proceedings of the 15th International Conference on Mathematical and Computational Methods in Science and Engineering* (*MACMESE '13*), Kuala Lumpur, Malaysia, 2013, pp. 56-61.

[23] P. Zítek, V. Kučera, and T. Vyhlídal, Meromorphic Observer-Based Pole Assignment in Time Delay Systems, *Kybernetika*, Vol.44, No.5, pp. 633-648, 2008.

[24] J.P. Richard, Time-Delay Systems: An Overview of Some Recent Advances and Open Problems, *Automatica*, Vol.39, No.10, 2003, pp. 1667-1694.

[25] O. Diekmann, S.A. Van Gils, S.M. Verduyn Lunel and H.-O. Walther, *Delay Equations: Functional, Complex and Nonlinear Analysis*, Applied Math. Sciences Vol.110, New York: Springer, 1995. vol. 110.

[26] D. Breda, Solution Operator Approximation for Characteristic Roots of Delay Differential Equations, *Applied Numerical Mathematics*, Vol.56, No.3-4, 2006, pp. 305-317.

[27] L. Pekař and R. Prokop, Algebraic Optimal Control in RMS Ring: A Case Study, *International Journal of Mathematics and Computers in Simulation*, Vol.7, No.1, 2013, pp. 59-68.